Package: rocnp (via r-universe)

November 8, 2026

Title Work with Romanian Personal Numeric Codes PNC / CNP

Version 0.1.0.9001

Description A set of tools for working with Romanian personal numeric codes. The core is a validation function which applies several verification criteria to assess the validity of numeric codes. This is accompanied by functionality for extracting the different components of a personal numeric code. A personal numeric code is issued to all Romanian residents either at birth or when they obtain a residence permit.

License MIT + file LICENSE

URL https://github.com/dragosmg/rocnp

BugReports https://github.com/dragosmg/rocnp/issues

Depends R (>= 4.1.0)

Imports cli, dplyr, glue, lubridate, purrr, rlang, stringr, tibble, vctrs

Suggests covr, testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Config/pak/sysreqs libicu-dev

Repository https://dragosmg.r-universe.dev

Date/Publication 2025-09-14 16:00:25 UTC

RemoteUrl https://github.com/dragosmg/rocnp

RemoteRef HEAD

RemoteSha e90530241013b1dece9f532e02a33ed089cb6061

Contents

cnp 2
county_lookup 3
days_in_month 3
extract_sex 4

Index **6**

---

cnp *Create a* cnp *object*

Description

- `cnp()` is used to build a cnp vector
- `is_cnp` is used to check whether a vector is of class cnp

Usage

```
cnp(x = character())

is_cnp(x)
```

Arguments

x	(character) a character (or values that can be cast to character) vector of personal numeric codes (CNPs).

Value

- `cnp()`: a vector of class cnp.
- `is_cnp()`: a logical vector.

Examples

```
a <- cnp(c("1940616346114", "7041218318525"))

is_cnp(a)

is_cnp(1L:3L)
```

county_lookup *County lookup table*

Description

A lookup table mapping the county code to the county name.

Usage

```
county_lookup
```

Format

`county_lookup`**:**
A data frame with 48 rows and 2 columns:

code county code
county county name

Source

https://ro.wikipedia.org/wiki/Cod_numeric_personal_(RomÃćnia)#JJ

days_in_month *Days in month lookup table*

Description

A lookup table containing the number of days in a month.

Usage

```
days_in_month
```

Format

`days_in_month`**:**
A data frame with 12 rows and 3 columns:

month month as padded character - `"01"` to `"12"`
days_non_leap number of days in a non-leap year
days_leap number of days in a leap year

`extract_sex` *Extract CNP components*

Description

The `extract_()` family of functions provides tools to extract the different pieces of information encoded in the CNP. These functions return a character vector:

- `extract_sex()` extracts the sex (`"M"` or `"F"`) based on the `S` component of the CNP.
- `extract_birth_year()` extracts the year of birth, inferred from the `S` (which also encodes the century of birth, alongside the sex) and `AA` (encoding the final 2 digits of the birth year) components of the CNP.
- `extract_birth_month()` extracts the month of birth based on the `"LL"` component.
- `extract_county()` gets the county of issue (based on the `JJ` component).
- `extract_status()` extracts the residence status (based on the `S` component)

This function returns a date vector:

- `extract_dob()` gets the date of birth (based on the `S`, `AA`, `LL`, and `ZZ` components).

Usage

```
extract_sex(x)

extract_birth_year(x)

extract_birth_month(x)

extract_dob(x)

extract_county(x)

extract_status(x)
```

Arguments

`x`	a `cnp` vector.

Value

- `extract_sex()`: a character vector containing the parsed sex: `M` or `F`.
- `extract_birth_year()`: a character vector containing the birth year.
- `extract_birth_month()`: a character vector containing the birth month.
- `extract_county()`: a character vector containing the county of issue.
- `extract_status()`: a character vector containing the residence status (`"native"` or `"resident"`).
- `extract_dob()`: a vector of containing the date of birth.

Examples

```
# Extract sex
extract_sex(cnp(c("5201206346491", "1940616346114", "7041218318525", NA)))

# Extract birth year
extract_birth_year(
  cnp(c("5201206346491", "1940616346114", "7041218318525", NA))
)

# Extract birth month
extract_birth_month(cnp(c(5201206346491, 1940616346114, 7041218318525, NA)))

# Extract date of birth
extract_dob(cnp(c("5201206346491", "1940616346114", "7041218318525", NA)))

# Extract county
extract_county(cnp(c("5201206346491", "1940616346114", "7041218318525", NA)))

# Extract residence status
extract_status(cnp(c("5201206346491", "1940616346114", "7041218318525", NA)))
```

Index

∗ **datasets**
county_lookup, 3
days_in_month, 3

cnp, 2
county_lookup, 3

days_in_month, 3

extract_birth_month *(*extract_sex*)*, 4
extract_birth_year *(*extract_sex*)*, 4
extract_county *(*extract_sex*)*, 4
extract_dob *(*extract_sex*)*, 4
extract_sex, 4
extract_status *(*extract_sex*)*, 4

is_cnp *(*cnp*)*, 2